Simulation and optimization of Cologne's tram schedule

Oliver Ullrich[a], Daniel Lückerath[a,b], Sebastian Franz[a],
Ewald Speckenmeyer[a]
[a]Institut für Informatik, Universität zu Köln
[b]Institut für Nachrichtentechnik, Fachhochschule Köln
ullrich|lueckerath|franz|esp@informatik.uni-koeln.de

Abstract

In many tram networks multiple lines share tracks and stations, thus requiring robust schedules which prevent inevitable delays from spreading through the network. Feasible schedules also have to fulfill various planning requirements originating from political and economical reasons.

In this paper we present a tool set designed to generate schedules optimized for robustness, which also satisfy given sets of planning requirements. These tools allow us to compare time tables with respect to their applicability and evaluate them prior to their implementation in the field.

This paper begins with a description of the tool set focusing on optimization and simulation modules. These software utilities are then employed to generate schedules for our hometown Cologne's tram network, and to subsequently compare them for their applicability.

1 Introduction

In many tram networks, several lines share resources like stations and tracks. This results in very dense schedules, with vehicles leaving platforms every minute at peak times. In order to prevent inevitable local delays from spreading through the network, a schedule has to be robust.

Many additional planning requirements to real world tram schedules originate from political, economical and feasibility reasons. Thus it is not sufficient to exclusively consider general criteria like robustness or operational costs when generating time tables. Typical requirements include fixed start times at certain stations, e.g. interfaces to national railway systems, core lines that relieve high passenger load, e.g. for lines which traverse city centers, warranted connections at certain stations, and safety distances to be complied with at bidirectional tracks.

In this paper we present an introduction to our project to generate and evaluate robust time tables which also satisfy given sets of planning requirements. We describe a tool chain which enables us to generate optimized schedules, compare their applicability and evaluate them prior to their implementation in the field.

This paper continues with a description of the current state of the project, focusing on our approaches on optimization and simulation (Section 2). We then present some experimental results obtained by applying the described software to our hometown Cologne's tram network (Section 3). The paper closes with a short summary of lessons learned and some thoughts on further research (Section 4).

2 Simulating and optimizing tram schedules

Our project *"Computer Aided Traffic Scheduling" (CATS)* is built around a database complying with the ÖPNV5 data model released by the *Association of German Transport Companies (Verband Deutscher Verkehrsunternehmen*, see [19]). Visualization, optimization, and simulation modules are connected via operations on the database and through XML configuration files (see figure 1). Due to its compliance with the ÖPNV5 data model our framework is capable of working on many European tram networks.

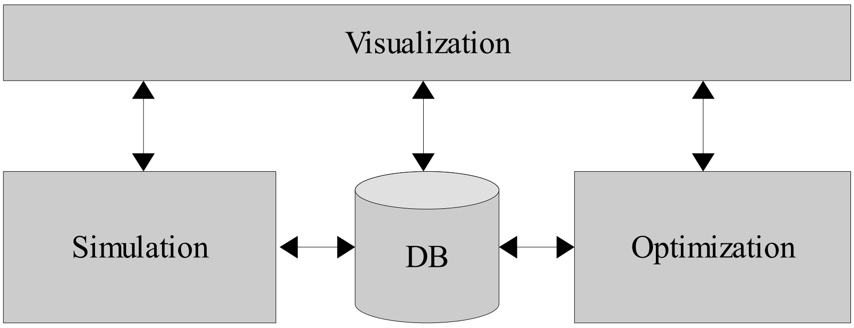

Figure 1: Project modules

2.1 Optimization of tram schedules

Various approaches to optimize tram and railway schedules are known (see e.g. [1, 3, 4, 7, 16, 17, 18]). Most of them aim at one general objective like minimizing vehicle delay (see [16, 18]) or maximizing robustness to restrict the global impact of small, local disturbances (see [4, 7]). Others use a combination of objectives, like operational profit and robustness in [3], or combining social opportunity cost and operational cost in [17].

Because of the complex nature of the problem, many authors use heuristic approaches like Lagrangian heuristics (see [3]) or simulated annealing (see [17]). Others, like Bampas et al. in [1] introduce exact algorithms for restricted subclasses, like chain and spider networks.

In our project, we combine heuristics and exact methods to generate optimal synchronized time tables for general tram networks, targeting maximal robustness and adherence to transport planning requirements at the same time.

To calculate the robustness of a time table we examine at each platform the safety distance $\delta_{f,pred(f)}$ of any trip f and its predecessor $pred(f)$, i.e. the time elapsed between the departures of $pred(f)$ and f at the examined station. To reduce complexity we aggregate subsequent similar platforms operated by the same lines to a maximal platform type h', weighted by the number of included platforms φ_h (see figure 2). The reduced set of platforms is denoted by H'.

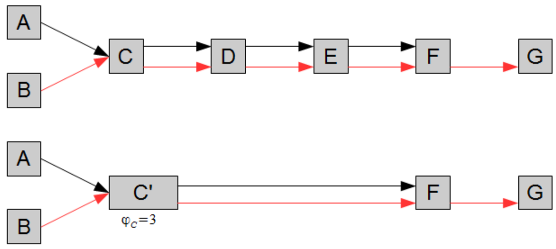

Figure 2: Example of platform reduction

To calculate the robustness $\Phi_a(\lambda)$ of schedule λ, we add the inverse of $\delta_{f,pred(f)}$ for all platforms and all trips, thus applying a penalty for small safety distances. With F_h representing all trips that serve platform h, the resulting function is as follows:

$$\Phi_a(\lambda)=\sum_{h\in H'}\sum_{f\in F_h}\frac{1}{\delta_{f,pred(f)}}*\varphi_h$$

In order to calculate the compliance with transport planning requirements we introduce $\rho_v\in\{1,2,3,\infty\}$, the compliance factor of requirement v. A compliance factor of 1 means that the requirement is completely satisfied, 2 and 3 denote tolerable compliance, and ∞ means that the constraint is not met and the time table candidate must be rejected. With V denoting the set of all planning requirements, we add the compliance values and get the following:

$$\Phi_b(\lambda)=\sum_{v\in V}\rho_v$$

Depending on the network under consideration and the number of planning requirements, the two parts of the objective function may not be comparable directly. Thus we define a normalizing factor σ, which reflects the relationship between the theoretically optimal safety distance $\delta^{opt}_{f,pred(f)}$, obtained by dividing the tact interval by the number of serving lines, and the optimal compliance factor ρ_v^{min}. We define σ as:

$$\sigma=\left(\sum_{h\in H'}\sum_{f\in F_h}\frac{1}{\delta^{opt}_{f,pred(f)}}*\varphi_h\right)/\sum_{v\in V}\rho_v^{min}$$

Combining $\Phi_a(\lambda)$ and $\Phi_b(\lambda)$ yields the overall objective function $\Phi(\lambda)$, normalized by σ and weighted by α, the relative weight of the fulfillment of planning requirements.

$$\Phi(\lambda)=(1-\alpha)*\sum_{h\in H'}\sum_{f\in F_h}\frac{1}{\delta_{f,pred(f)}}*\varphi_h+\alpha*\sum_{v\in V}\rho_v*\sigma \quad \text{with } \alpha\in[0,1)$$

We identify seven types of transport planning constraints: Interval constraints, start time constraints, core line constraints, bidirectional track constraints, turning point constraints, warranted connection constraints and follow-up connection constraints.

Upon closer inspection it becomes clear that interval and start time constraints are elemental and all other constraint types can be expressed using these two. E.g. a bidirectional track constraint can be expressed by two interval constraints covering opposite platforms. Subsequently only interval and start time constraints are considered in the remainder of this paper.

To accelerate the computational process the implemented branch-and-bound solver is preceded by a genetic algorithm. We encode a time table into one individual, consisting of the first trip start time of each line, i.e. the offset in minutes from the start of the operational day. All other trips follow determined by their line's tact interval. The application generates a start population using random start time values, testing validity against planning constraints and collisions on network nodes. To reduce computational complexity we apply simple deterministic tournament selection and two-point-crossover (as described in [5]). After evaluation of several mutation methods, including random, minimal, and maximum enhancement mutation we choose a minimal random mutation method that only allows start times to be altered by one minute. We utilize a steady state replacement method, also described in [5]. At the end of each run a hill climbing algorithm is applied to the best individual to further improve its fitness.

As described above we use the best individual encountered by the genetic algorithm to provide the branch-and-bound solver with an initial upper bound, thus avoiding a cold start. Each inner node of the search tree represents a partial solution of the problem (see [8]). The root of the tree corresponds to a solution in which no line's start time is fixed. With each level of the tree admissible start times for an additional line are set.

The objective function is modified in order to cut branches off the tree as soon as possible. The set of lines L is divided into subsets of lines that are already fixed $\hat{L}$ and lines that are not yet fixed $\tilde{L}$. Accordingly we divide the set of transport planning constraints V into $\hat{V}$ and $\tilde{V}$ as well as the set of platforms H into $\hat{H}$ and $\tilde{H}$. The modified objective function $\Phi'(\lambda)$ is shown below.

$$\Phi'(\lambda)=(1-\alpha)*(\sum_{h\in\hat{H}}\sum_{f\in F_h}\frac{1}{\delta_{f,pred(f)}}*\varphi_h+\sum_{h\in\tilde{H}}\sum_{f\in F_h}\frac{1}{\tilde{\delta}_{f,pred(f)}}*\varphi_h)+\alpha*(\sum_{v\in\hat{V}}\rho_v+\sum_{v\in\tilde{V}}\rho_v^{min})*\sigma$$

$\tilde{\delta}_{f,pred(f)}$ represents the theoretically best safety distance value under consideration of lines already fixed. Again, ρ_v^{min} denotes the optimal compliance factor for constraint v. These values are applied in order to find a lower bound for solution candidates in the current branch of the search tree.

For further implementation details, see [6].

2.2 Simulation of tram schedules

Most rail-bound traffic simulations are designed for long distance train or railway networks, see e.g. [13, 15]. While those systems feature similarities to tram networks, e.g. passenger exchange or maneuvering capabilities, they differ greatly in important aspects. Tram networks are often mixed, i.e. trams travel on underground tracks as well as on street level, and are thus subject to individual traffic and corresponding traffic regulation strategies. Subsequently, tram behavior is a mixture between train and car behavior, e.g. line-of-sight operating/driving. Therefore a simple adaption of railway simulation methodologies is not feasible.

Bearing the similarities with individual traffic in mind Joisten implemented an adapted Nagel/Schreckenberg model (see [14]) for tram simulation, which suffered from the setbacks of the high aggregation inherent to cellular automatons (see [10]). Therefore Lückemeyer developed an event based simulation model which avoids some of those setbacks as described in [9, 10]. To further eliminate inaccuracies we apply an updated model, as described in [12].

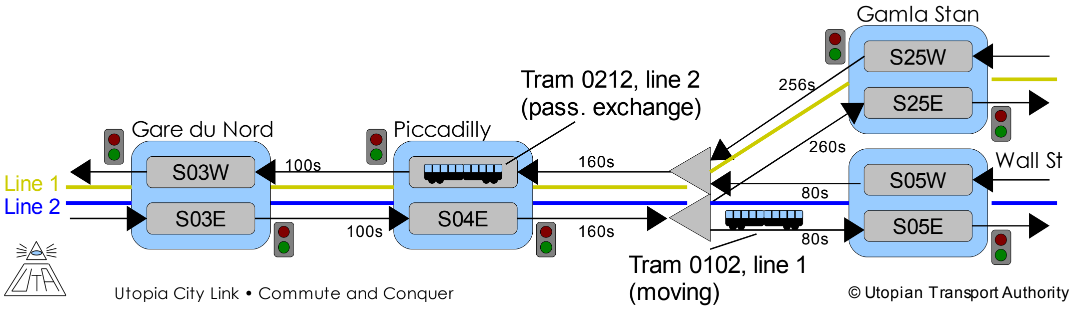

Figure 3: Part of a tram network

Our application is based upon a model-based parallelization framework, which exploits the embedded model's intrinsic parallelism. The mixed tram network is modeled as a directed graph with platforms, tracks and track switches represented by nodes. Connections between nodes are represented as edges. Figure 3 shows part of an example network, which is mapped on the graph depicted in figure 4, where squares represent platforms, rectangles tracks and triangles track switches. The rectangles around platforms indicate that these platforms form a station.

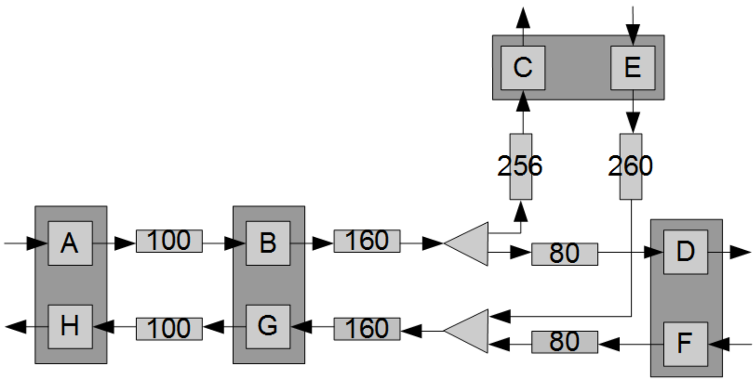

Figure 4: Example graph representing part of a tram network

Passenger boarding and deboarding time distributions are specific to platform and tram type with the combined duration of opening and closing the vehicle doors as minimum value.

Vehicles encapsulate most of the simulation dynamics, which are based upon the event based simulation approach (as described in [2]). Thus trams change their state at events of certain types, like stopping or opening doors, which happen at discrete points in time. These state changes may trigger a change in the overall system state and generate follow-up events, which are administrated in a priority queue.

Main tram attributes are specified by the type of tram, which holds functions for the maneuvering capabilities, e.g. acceleration and braking.

For further implementation details, see [11] and [12].

3 Experiments

3.1 Optimizing Cologne's tram network

We apply the developed software suite to our hometown Cologne's tram network based on the time table data of 2001 (see figure 5). It consists of 528 platforms and 58 track switches connected via 584 tracks. These tracks cover a total length of 407.4 kilometers, resulting in an average track length of 697.6 meters. 15 lines with 182 line routes are served by 178 vehicles which execute 2,814 trips per operational day.

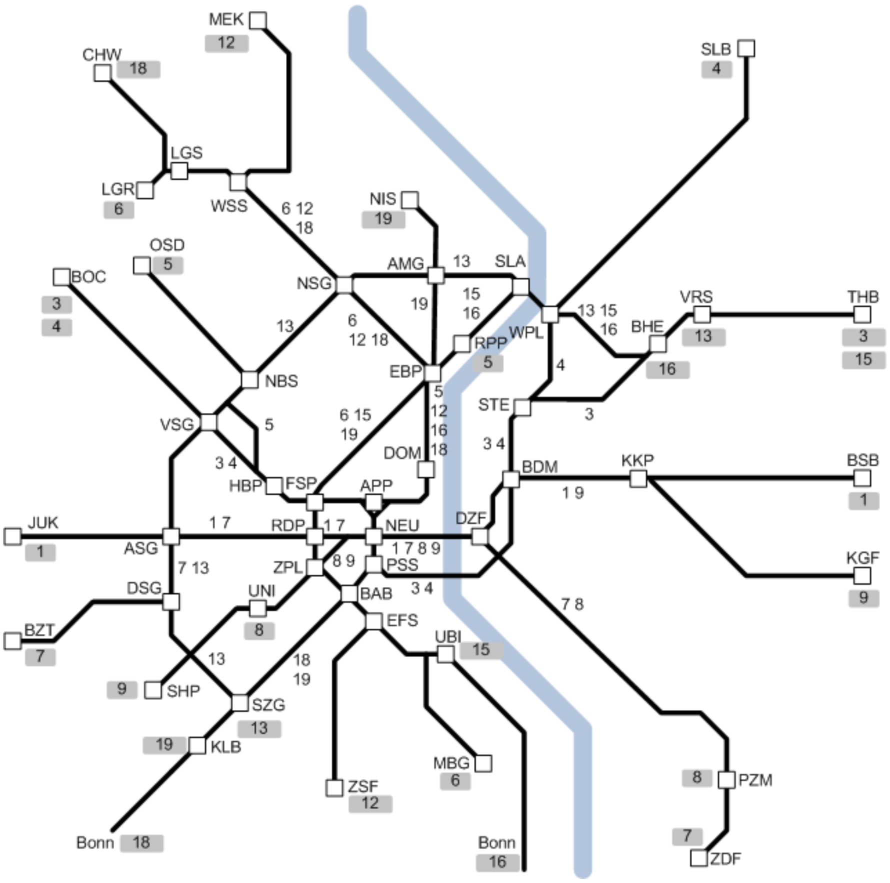

Figure 5: Cologne's tram network in 2001

For optimization purposes, we only consider the 36 major routes. The remaining 146 minor routes are usually trips between the start or end point of a regular trip and depots, or other maintenance trips at the rim of the network. For the following optimization run, we assume a tact interval of ten minutes, and define a set of example constraints, which can be decomposed to four start time constraints and 34 interval constraints. These include minimum turn-around times at line ends, an additional core line 1A to satisfy high demand for line 1 in Cologne's town center, guaranteed connections between certain lines, and fixed start times at the Bonn national railway hub.

3.2 Comparing two tram schedules

From the genetic algorithm's initial pool of valid solution candidates we randomly take a schedule A with an objective function value of 7,655.14 (see table 1). After a 166 minutes run, the optimizer yields a best solution candidate B with an objective function value of 6,786.60 (see table 2).

Direction	**Line**														
	1	1A	3	4	5	6	7	8	9	12	13	15	16	18	19
Forward	4	5	9	4	0	3	5	3	7	2	9	4	6	9	8
Backward	5	2	8	6	9	8	1	0	6	0	4	5	5	7	3

Table 1: Schedule A – Initial schedule

We examine both schedules by executing 30 simulation runs and comparing the results. Schedule A yields an average line delay of 23.6 seconds, while schedule B yields one of 17.7 seconds. As seen in figure 6, implementation of schedule B enhances punctuality of every line at least marginally. Lines 6, 7 and 18 in particular are improved significantly, reducing line delay between 20 and 40 percent. Lines 1, 8 and 13 feature an even more improved punctuality (see table 3) and thus deserve a closer examination.

Direction	**Line**														
	1	1A	3	4	5	6	7	8	9	12	13	15	16	18	19
Forward	8	4	3	9	5	4	9	4	2	5	7	5	6	9	7
Backward	4	2	4	1	3	8	0	1	7	8	7	8	5	7	2

Table 2: Schedule B – Best schedule

Line 1 (combined with line 1A) traverses the highly frequented city center every 5 minutes and shares important resources with lines 7, 8 and 9. Thus it is very susceptible to small disturbances originating in those highly requested areas. In comparison to schedule A, schedule B's better utilization of safety distances improves punctuality by 43 percent.

Outside the town center, line 8 yields a particularly high delay under schedule A due to a marginal safety distance between its vehicles and those of its immediate predecessor line 7. Therefore trams of line 8 are prone to resource conflicts with vehicles of that line. In schedule B the resulting delay is reduced to 28 percent by increasing the safety distances to 4 and 6 minutes respectively.

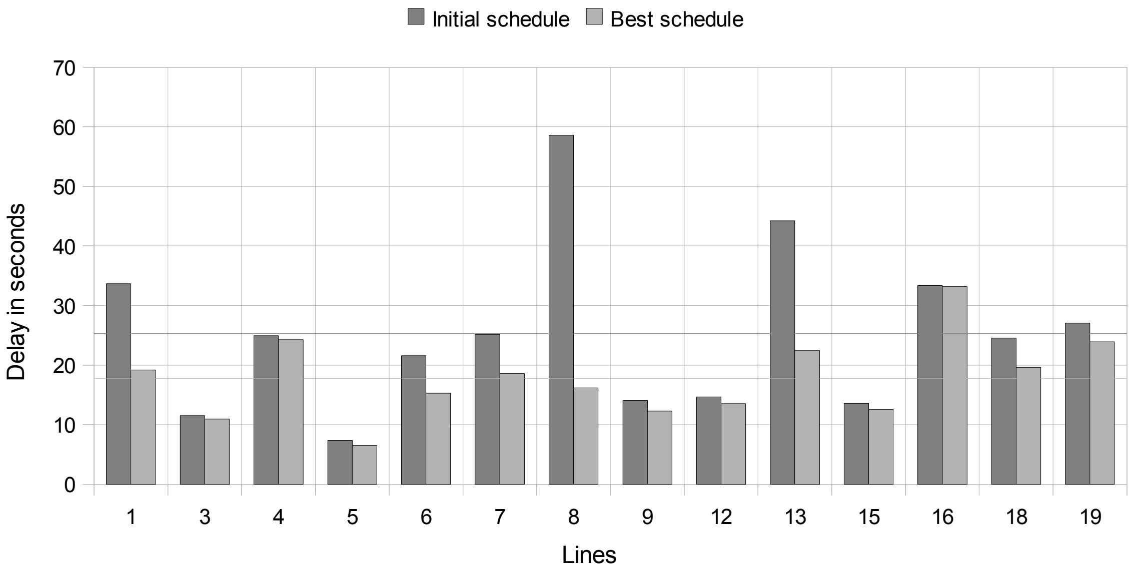

Figure 6: Average delay of lines

Examining the planned departure times of north-east bound line 13 under time table A shows that even small delays resulting from conflicts with lines 5 or 7 cause vehicles of the line to fall directly behind those of line 15. This further prolongs their delays and makes it impossible to catch up on pre-existing delays. Also under schedule A, south-west moving trams of line 13 are placed directly behind vehicles of lines 15 and 16, thus resulting in a high receptiveness for delay. Schedule B resolves those issues, resulting in a decrease in delay of 49 percent.

Line		**1**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**12**	**13**	**15**	**16**	**18**	**19**
Ø Delay	A	33,7	11,5	24,9	7,3	21,6	25,2	58,6	14,1	14,7	44,2	13,6	33,3	24,5	27,1
	B	19,2	10,9	24,3	6,5	15,3	18,6	16,2	12,3	13,5	22,4	12,5	33,2	19,6	23,9
Abs. gain		14,5	0,6	0,6	0,8	6,3	6,6	42,4	1,8	1,2	21,6	1,1	0,1	4,9	3,2
Rel. gain		0,43	0,05	0,02	0,11	0,29	0,26	0,72	0,13	0,08	0,49	0,08	0,00	0,20	0,12

Table 3: Comparing schedules: Lines

Simulation data collected at the important hubs Barbarossaplatz (BAB-1 to BAB-4), Ebertplatz (EBP-1 to EBP-4), and Neumarkt (NEU-1 to NEU-4) is presented in table 4. Under schedule B, delay was reduced significantly at eight of those platforms, staying on about the same level at further two (see figure 7). The increase in punctuality can be explained by the better reliability of the frequenting lines under the optimized schedule.

The rise in delay at platforms EBP-3 (8.5 seconds) and NEU-4 (2.3 seconds) remains to be explained. Both platforms are preceded by highly frequented tracks, used by lines whose punctuality does not improve significantly by applying schedule B. This would partly explain the average delay to be stagnant. Furthermore, these tracks are merged by arrays of underground track switches, which have to be negotiated by every incoming vehicle. The timing changes from schedule A to schedule B could yield adverse configurations of switch tongues, which would explain the observed small increase in delay.

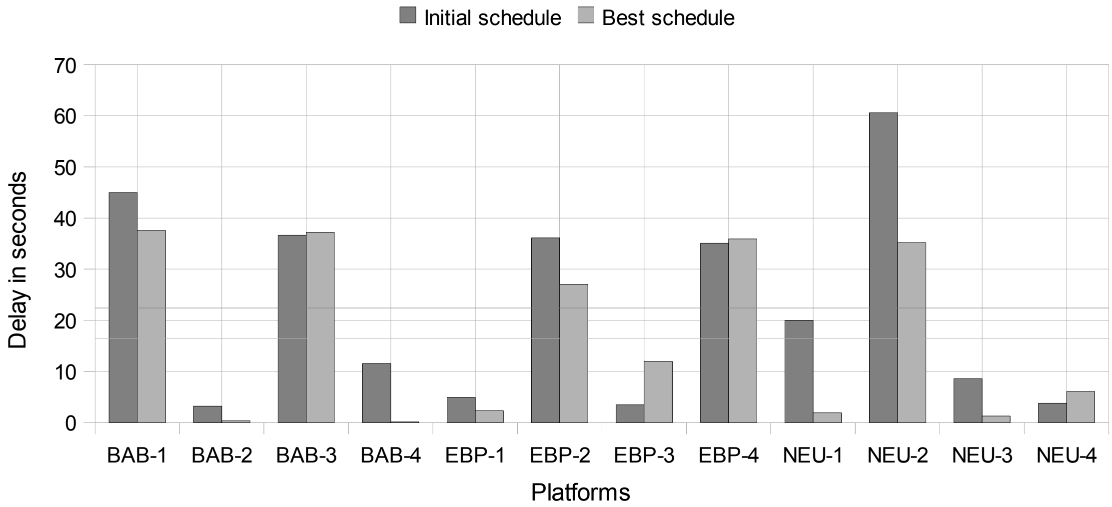

Figure 7: Average delay at platforms

Platform		**BAB-1**	**BAB-2**	**BAB-3**	**BAB-4**	**EBP-1**	**EBP-2**	**EBP-3**	**EBP-4**	**NEU-1**	**NEU-2**	**NEU-3**	**NEU-4**
Ø Delay	A	45,0	3,2	36,7	11,6	5,0	36,1	3,5	35,1	20,0	60,6	8,6	3,8
	B	37,6	0,4	37,2	0,1	2,3	27,1	12,0	35,9	1,9	35,2	1,3	6,1
Abs. gain		7,4	2,9	-0,6	11,5	2,6	9,1	-8,5	-0,8	18,1	25,4	7,3	-2,3
Rel. gain		0,16	0,89	-0,01	0,99	0,53	0,25	-2,41	-0,02	0,9	0,42	0,85	-0,61

Table 4: Comparing schedules: Platforms

4 Conclusion and future work

In this paper we presented a tool chain to generate and evaluate tram schedules. The described optimization module is capable of generating robust time tables which fulfill planning requirements as found in real world projects. We also presented a simulation engine which makes it possible to test real and generated schedules for their applicability and so to further validate them.

We applied the described tool chain to our hometown Cologne's mixed tram network. A random but valid time table A was compared to the resulting best schedule B. As to be expected, the average delay under schedule B is significantly lower than that under schedule A. All lines gain punctuality, though at some core platforms the average delay rises for up to nine seconds.

In further steps more detailed studies of tram networks and schedules will be carried out, including Cologne's new underground tracks currently under construction, which are designed to relieve the central Neumarkt tunnel. We found it desirable to be able to manually apply small incremental changes to a schedule while getting instant visual assessment of expected consequences. A tool with those capabilities is in the planning stage. Furthermore the optimizer module will be parallelized to further reduce its run time. Especially the applied branch-and-bound algorithm's load can be balanced relatively easy, so the application should scale well.

Acknowledgements

This work was partially supported by *Rhein Energie Stiftung Jugend Beruf Wissenschaft* under grant number W-10-2-002. We thank the other members of our work group Patrick Kuckertz and Bert Randerath.

References

[1] Bampas, E., Kaouri, G., Lampis, M., Pagourtzis, A.: *Periodic Metro Scheduling*. In: Proceedings of ATMOS, 2006

[2] Banks, J., Carson, J.S., Nelson B.L., Nicol D.M.: *Discrete-Event System Simulation.* Pearson, 2010

[3] Cacchiana, V., Caprara, A., Fischetti, M.: *A Lagrangian Heuristic for Robustness, with an Application to Train Timetabling*. Transportation Science, to appear

[4] Caimi, G., Fuchsberger, M., Laumanns, M., Schüpbach, K.: *Periodic Railway Timetabling with Event Flexibility*. In: Networks, 2010, Volume 57, Number 1, pp. 3-18

[5] Dréo, J., Pétrowski, A., Siarry, P., Taillard, E.: *Metaheuristics for Hard Optimization.* Springer, 2006

[6] Franz, S.: *Entwurf und Entwicklung eines mehrstufigen Optimierungsverfahrens für Stadtbahnfahrpläne unter Berücksichtigung verkehrsplanerischen Vorgaben*. Diplomarbeit, Univ. Köln, 2011

[7] Genç, Z.: *Ein neuer Ansatz zur Fahrplanoptimierung im ÖPNV: Maximierung von zeitlichen Sicherheitabständen*. Dissertation, Mathematisch-Naturwissenschaftliche Fakultät, Universität zu Köln, 2003

[8] Hu, T.C.: *Combinatorial Algorithms*, Addison Wesley, 1982

[9] Lückemeyer, G.: *A Traffic Simulation System Increasing the Efficiency of Schedule Design for Public Transport Systems Based on Scarce Data.* Dissertation, Shaker Verlag, 2007

[10] Lückemeyer, G., Speckenmeyer, E.: *Comparing Applicability of Two Simulation Models in Public Transport Simulation.* In: Becker, M., Szczerbicka, H. (Ed.): Proceedings of ASIM 2006, Hannover, 2006

[11] Lückerath, D.: *Entwurf und Entwicklung einer Anwendung zur parallelen Simulation von schienengebundenem Öffentlichen Personennahverkehr*. Diplomarbeit, Univ. Köln, 2011

[12] Lückerath, D., Ullrich, O., Speckenmeyer, E.: *Modeling time table based tram traffic.* In: Bödi, R., Maurer, W. (Ed.): Proceedings of ASIM 2011, Winterthur, 2011

[13] Middelkoop, D., Bouwman, M.: *SIMONE: Large Scale Train Network Simulations*. In: B.A. Peters, J.S. Smith, D.J. Medeiros, M.W. Rohrer (Ed.): Proceedings of the 2011 Winter Simulation Conference, Arlington, 2001

[14] Nagel, K., Schreckenberg, M.: *A cellular automaton model for freeway traffic*. In: Journal de Physique I, Volume 2, Issue 12, December 1992, pp. 2221-2229

[15] Nash, A., Huerlimann, D.: *Railroad Simulation Using OpenTrack*. In: Allan, J., R.J. Hill, C.A. Brebbia, G. Sciutto, S. Sone (Ed.): Computers in Railways IX, WIT Press, Southampton, 2004, pp. 45-54

[16] Schöbel, A.: *A Model for the Delay Management Problem based on Mixed-Integer-Programming*. In: Proceedings of ATMOS, 2001

[17] Speckenmeyer, E., Li, N., Lückerath, D., Ullrich, O.: *Socio-Economic Objectives in Tram Scheduling*. Technical Report, Universität zu Köln, to appear

[18] Suhl, L., Mellouli, T.: *Managing and preventing delays in railway traffic by simulation and optimization*. In: Mathematical Methods on Optimization in Transportation Systems 2001, pp. 3–16

[19] Verband Deutscher Verkehrsunternehmen e.V.: *VDV-Standardschnittstelle Liniennetz/Fahrplan,* VDV-Schriften 452, 2008